

AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

- 1 1. (Currently amended) A method for scheduling processes within an
2 operating system based upon virtual server identifiers, wherein the operating
3 system supports multiple virtual servers that operate within separate virtual
4 environments on a single computing platform, the method comprising:
5 | the operating system detecting an event that causes a scheduling priority
6 | for a process to be updated;
7 | the operating system looking up a virtual server identifier for the process,
8 | wherein the virtual server identifier specifies a virtual server and an associated
9 | virtual environment that the process operates within;
10 | the operating system using the virtual server identifier to look up a
11 | scheduling priority associated with the virtual server; and
12 | the operating system calculating an updated scheduling priority for the
13 | process based upon the scheduling priority associated with the virtual server;
14 | wherein calculating the updated scheduling priority involves calculating
15 | the updated scheduling priority based upon:
16 | a value, E , stored within a priority-related timer that keeps
17 | track of execution time for the process,
18 | a system priority, S_p , associated with the process, and
19 | the scheduling priority, M , associated with the virtual
20 | server; and

21 | wherein calculating the updated scheduling priority, P , involves
22 | calculating $P = S_p + S(E/M)$, wherein S is a tunable constant value.

1 2. (Canceled).

1 3. (Canceled).

1 4. (Original) The method of claim 1, wherein the method further
2 comprises:
3 receiving a command to adjust the scheduling priority associated with the
4 virtual server;
5 if the command is received from an authorized entity, adjusting the
6 scheduling priority associated with the virtual server so that the scheduling
7 priorities of all processes associated with the virtual server are modified.

1 5. (Original) The method of claim 1, wherein the method further
2 comprises charging a fee for hosting the virtual server, wherein the fee is based
3 upon the scheduling priority associated with the virtual server.

1 6. (Original) The method of claim 1, wherein detecting the event that
2 causes the scheduling priority for the process to be updated involves detecting one
3 of:
4 the process entering a sleep state;
5 the process waking up from the sleep state; and
6 a priority-related timer associated with the process reaching a maximum
7 value.

1 7. (Original) The method of claim 1, wherein looking up the virtual
2 server identifier for the process involves looking up the virtual server identifier
3 within a process structure maintained by the operating system for the process.

1 8. (Currently amended) A computer-readable storage medium storing
2 instructions that when executed by a computer cause the computer to perform a
3 method for scheduling processes within an operating system based upon virtual
4 server identifiers, wherein the operating system supports multiple virtual servers
5 that operate within separate virtual environments on a single computing platform,
6 the method comprising:

7 | the operating system detecting an event that causes a scheduling priority
8 for a process to be updated;

9 | the operating system looking up a virtual server identifier for the process,
10 wherein the virtual server identifier specifies a virtual server and an associated
11 virtual environment that the process operates within;

12 | the operating system using the virtual server identifier to look up a
13 scheduling priority associated with the virtual server; and

14 | the operating system calculating an updated scheduling priority for the
15 process based upon the scheduling priority associated with the virtual server;

16 | wherein calculating the updated scheduling priority involves calculating
17 the updated scheduling priority based upon:

18 a value, E , stored within a priority-related timer that keeps
19 track of execution time for the process.

20 a system priority, S_P , associated with the process, and
21 the scheduling priority, M , associated with the virtual
22 server; and

23 | wherein calculating the updated scheduling priority, P , involves
24 calculating $P = S_P + S(E/M)$, wherein S is a tunable constant value.

1 9. (Canceled).

1 10. (Canceled).

1 11. (Original) The computer-readable storage medium of claim 8,
2 wherein the method further comprises:
3 receiving a command to adjust the scheduling priority associated with the
4 virtual server;
5 if the command is received from an authorized entity, adjusting the
6 scheduling priority associated with the virtual server so that the scheduling
7 priorities of all processes associated with the virtual server are modified.

1 12. (Original) The computer-readable storage medium of claim 8,
2 wherein the method further comprises charging a fee for hosting the virtual server,
3 wherein the fee is based upon the scheduling priority associated with the virtual
4 server.

1 13. (Original) The computer-readable storage medium of claim 8,
2 wherein detecting the event that causes the scheduling priority for the process to
3 be updated involves detecting one of:
4 the process entering a sleep state;
5 the process waking up from the sleep state; and
6 a priority-related timer associated with the process reaching a maximum
7 value.

1 14. (Original) The computer-readable storage medium of claim 8,
2 wherein looking up the virtual server identifier for the process involves looking up

3 the virtual server identifier within a process structure maintained by the operating
4 system for the process.

1 15. (Currently amended) An apparatus that schedules processes within
2 an operating system based upon virtual server identifiers, wherein the operating
3 system supports multiple virtual servers that operate within separate virtual
4 environments on a single computing platform, the apparatus comprising:

5 | a detection mechanism in the operating system that is configured to detect
6 an event that causes a scheduling priority for a process to be updated;

7 | a lookup mechanism in the operating system that is configured to look up
8 a virtual server identifier for the process, wherein the virtual server identifier
9 specifies a virtual server and an associated virtual environment that the process
10 operates within;

11 wherein the lookup mechanism is additionally configured to use the virtual
12 server identifier to look up a scheduling priority associated with the virtual server;
13 and

14 | a calculating mechanism in the operating system that is configured to
15 calculate an updated scheduling priority for the process based upon the scheduling
16 priority associated with the virtual server;

17 | wherein the calculating mechanism is configured to calculate the updated
18 scheduling priority based upon:

19 | a value, E , stored within a priority-related timer that keeps
20 track of execution time for the process.

21 | a system priority, S_p , associated with the process, and
22 the scheduling priority, M , associated with the virtual
23 server; and

24 wherein the calculating mechanism is configured to calculate the updated
25 scheduling priority, P , by calculating $P = S_P + S(E/M)$, wherein S is a tunable
26 constant value.

1 16. (Canceled).

1 17. (Canceled).

1 18. (Original) The apparatus of claim 15, further comprising a priority
2 adjustment mechanism that is configured to:
3 receive a command to adjust the scheduling priority associated with the
4 virtual server; and to
5 adjust the scheduling priority associated with the virtual server so that the
6 scheduling priorities of all processes associated with the virtual server are
7 modified, if the command is received from an authorized entity.

1 19. (Original) The apparatus of claim 15, further comprising a fee
2 calculation mechanism that is configured to calculate a fee for hosting the virtual
3 server based upon the scheduling priority associated with the virtual server.

1 20. (Original) The apparatus of claim 15, wherein the detection
2 mechanism is configured to detect one of:
3 the process entering a sleep state;
4 the process waking up from the sleep state; and
5 a priority-related timer associated with the process reaching a maximum
6 value.

1 21. (Original) The apparatus of claim 15, wherein the lookup
2 mechanism is configured to look up the virtual server identifier for the process by
3 looking up the virtual server identifier within a process structure maintained by
4 the operating system for the process.